NASA Procedural Requirements

NPR 7150.2
Effective Date: September 27, 2004
Expiration Date: September 27, 2009

COMPLIANCE IS MANDATORY

Printable Format (PDF)

Subject: NASA Software Engineering Requirements

Responsible Office: Office of the Chief Engineer

CHAPTER 2: Software Management Requirements

The software management activities define and control the many software aspects of a project from beginning to end. This includes the interfaces to other organizations, determination of deliverables, estimates and tracking of schedule and cost, risk management, formal and informal reviews as well as other forms of verification and validation, and determination of the amount of supporting services. The planned management of these activities is captured in one or more software and/or system plans.

2.1 Compliance with Laws, Policies, and Requirements

The software management process requires the understanding and application of other NASA policy requirements that impact the development, release, and/or maintenance of the software.

2.1.1 The project shall ensure that software disclosure requirements of NPD 2091.1, Inventions Made By Government Employees, are implemented by their project, Section 305 of the Space Act (42 U.S.C 2457) for large business contractors, and 35 U.S.C. 200 et seq., (including Section 202(c)) for small businesses, universities, and non-profits are implemented by their project. [SWE-007]

2.1.2 The project shall ensure that software technology transfer requirements of NPR 2190.1, NASA Export Control Program, are implemented by the project. [SWE-008]

2.1.3 The project shall ensure that software external release requirements of NPR 2210.1, External Release of NASA Software, are implemented by the project. [SWE-009]

2.1.4 The project shall ensure that the security requirements of NPD 2810.1, NASA Information Security Policy, are implemented by the project. [SWE-010]

2.1.5 The project shall ensure that the requirements of reasonable accommodation for individuals with disabilities per NPR 3713.1, Procedures for Providing Reasonable Accommodation for Individuals with Disabilities, are implemented by the project. [SWE-011]

2.1.6 The project shall ensure that software is accessible to individuals with disabilities as required by Section 508 of the Rehabilitation Act (29 U.S.C. 749d), as amended. Specific requirements for accessibility may be found at 36 CFR Part 1194. [SWE-012]

2.2 Software Life Cycle Planning

Software Life Cycle Planning covers the software aspects of a project from inception through retirement. It is meant as an organizing process that considers the software as a whole and provides the planning activities required to insure a coordinated, well-engineered process for defining and implementing project activities. These processes, plans, and activities are coordinated within the greater project. At project conception, software needs for the project are analyzed, including acquisition, supply, development, operation, maintenance, and supporting activities and processes. The software effort is scoped and the processes, measurements, and activities are documented in software plan(s).

2.2.1 The project shall develop software plan(s). [SWE-013]
Note: The requirement for the content of each software plan (whether stand-alone or condensed into one or more project level or software documents) is defined in Chapter 5. These include, but are not limited to:
a. Software development or management plan.
b. Software configuration management plan.
c. Software test plans.
d. Software maintenance plans.
e. Software assurance plans.

2.2.2 The project shall implement and execute the software plan(s). [SWE-014]

2.2.3 The project shall establish, document, and maintain at least one software cost estimate that satisfies the following conditions: [SWE-015]
a. Covers the entire software life cycle.
b. Is based on selected project attributes (e.g., assessment of the size, functionality, complexity, criticality, and risk of the software processes and products).
c. Is based on an assessment of the technology to be used and the impact on risk, cost, and schedule.

2.2.4 The project shall document and maintain a software schedule that satisfies the following conditions: [SWE-016]
a. Coordinates with the overall project schedule.
b. Documents the interactions of milestones and deliverables between software, hardware, operations, and the rest of the system.

2.2.5 The project shall plan, track, and ensure project specific software training for project personnel. [SWE-017]

2.2.6 The project shall regularly hold reviews of software activities, status, and results with the project stakeholders and track issues to resolution. [SWE-018]

2.2.7 The project shall select and document a software development life cycle or model that includes phase transition criteria for each life cycle phase (e.g., formal review milestones, informal reviews, software requirements review (SRR), preliminary design review (PDR), critical design review (CDR), test readiness reviews, customer acceptance or approval reviews). [SWE-019]

2.2.8 The project shall classify each of the systems and subsystems containing software in accordance with the software classifications definitions for Class A, B, C, D, E, F, G and H in Appendix B. [SWE-020]

Note: These classifications are documented in the Software Development or Management Plan as defined in Chapter 5. Software Assurance also performs an independent classification assessment and the results should be compared as per NASA-STD-8739.8, NASA Software Assurance.

2.2.9 If a system or subsystem evolves to a higher software classification as defined in Appendix B, then the project shall update its plan to fulfill the added requirements per the Requirements Mapping Matrix. [SWE-021]

2.2.10 The project shall ensure that software assurance is implemented by their project as per NASA-STD-8739.8, NASA Software Assurance. [SWE-022]
Note: Software assurance activities occur throughout the life of the project and, while some of the actual analyses may be performed within the project, NASA's safety and mission assurance organizations provide assurance that the products and processes are implemented according to the agreed upon plan(s). It is important to have software assurance of all software activities and products including Request for Proposals, contracts and memorandums of agreement, software plans, requirements, design, implementation, verification, validation, certification, acceptance, maintenance, operations, and retirement.

2.2.11 When a project is determined to have safety critical software, the project shall ensure that the safety requirements of NASA-STD-8719.13, Software Safety, are implemented by the project. [SWE-023]

2.2.12 The project shall ensure that actual results and performance are tracked against the software plans. [SWE-024]

2.2.13 The project shall ensure that corrective actions are taken and managed to closure when actual results and performance deviate from the software plans. [SWE-025]

2.2.14 The project shall ensure that changes to commitments (e.g., software plans) are agreed to by the affected groups and individuals. [SWE-026]

2.3 Commercial, Government, and Modified Off-The-Shelf Software

Since many of NASA's projects now contain off-the-shelf software products, it is important to plan and manage when and how to incorporate them. The off-the-shelf software discussed here apply only when those off-the-shelf software are to be included as part of a NASA system (per section P.2.1). The following requirements do not apply to

standalone desktop applications (e.g. MS Word, Adobe Acrobat, MS Power Point). However, when applications such as MS Excel or MS Access are used within a NASA system/subsystem as defined by the classes in this NPR, then they will need to be assessed and classified as part of the software subsystem in which they reside.

2.3.1 The project shall ensure that when COTS, GOTS, MOTS, open source, reuse, legacy, or heritage software product is to be acquired, the following conditions are satisfied: [SWE-027]
a. The requirements that are to be met by the off-the-shelf software are identified.
b. The off-the-shelf software includes documentation to fulfill its intended purpose (e.g. usage instructions).
c. Proprietary, usage, ownership, warranty, licensing rights, and transfer are addressed.
d. Future support for the off-the-shelf software product is planned.
e. Off-the-shelf software is validated to the same level of confidence as would be required of the developed software.

Note: It is the responsibility of the organization proposing to procure off-the-shelf software to document, prior to procurement, the plan for validating that such software can be assigned the same level of confidence that would be needed for an equivalent class of software if obtained through a "development" process.

Note: For critical systems or systems which must be maintained for long periods of time beyond the time a supplier would maintain or support the software the following should be considered:
a. Supplier agreement to deliver or escrow source code or third party maintenance agreement is in place.
b. A risk mitigation plan to cover the following cases is available:
(1) Loss of supplier or third party support for the product.
(2) Loss of maintenance for the product (or product version).
(3) Loss of the product (e.g., license revoked, recall of product, etc.)
c. Agreement that the project has access to defects discovered by the community of users has been obtained. When available, the project can join a product users group to obtain this information.
d. The plan to provide adequate support is in place, including timely maintenance and cost of maintenance.
e. Any changes to the software management, development, operations, or maintenance plans that are affected by the use or incorporation of COTS, GOTS, MOTS, reuse, legacy, or heritage software should be documented by the project.

2.4 Software Verification and Validation

Ensuring that the software products meet their requirements and that the products were built correctly is the purpose of verification and validation. Both software validation and software verification activities span the entire software life cycle and need to be planned. Formal and informal reviews, peer reviews/inspections, testing, demonstration, and analyses all can be used. Each project is generally free to choose the extent and combination of verification and validation methods and activities that best suit the project. Because peer reviews are such an important verification tool with proven value, there are specific peer review requirements in this NPR (Chapter 4).

2.4.1 The project shall plan software verification activities, methods, environments, and criteria for the project. [SWE-028]

Note: Software verification is a software engineering activity that demonstrates the software products meet specified requirements. Methods of software verification include: peer reviews/inspections of software engineering products for discovery of defects, software verification of requirements by use of simulations, black box and white box testing techniques, analyses of requirement implementation, and software product demonstrations. Planning for software verification should address the development, management review, and documentation for the software products. Refer to the Software Development or Management Plan software documentation requirement for software verification planning and incorporation (Chapter 5).

2.4.2 The project shall plan the software validation activities, methods, environments, and criteria for the project. [SWE-029]

Note: Software validation is a software engineering activity that demonstrates the as-built software product or software product component satisfies its intended use in its intended environment. Methods of software validation include: peer reviews/inspections of software product component behavior in a simulated environment, acceptance testing against mathematical models, analyses, and operational environment demonstrations. Planning for software validation should address the development, maintenance, support, and training for the software product and software product components. Refer to the Software Development or Management Plan software documentation requirement for software validation planning and incorporation (Chapter 5).

2.4.3 The project shall record, address, and track to closure the results of software verification activities. [SWE-030]

2.4.4 The project shall record, address, and track to closure the results of software validation activities. [SWE-031]

2.5 Project Formulation Requirements

Much of the project preparation and planning takes place during project formulation. While in the past, software planning was often left until later in the project life cycle, it is now seen as an essential part of the early planning phases and must be started as the project begins. This is especially important as software requirements must be properly incorporated into the project cost and schedule estimates, work planning, Request for Proposals, the evaluation of the contractors, and the contracts themselves.

2.5.1 Consistent with the Requirements Mapping Matrix (Appendix D), the project shall ensure that software is developed by either a software CMM® Maturity Level 3 or higher organization; or by an organization that has a CMMI®-SE/SW Capability Level 2 or higher as measured by a Software Engineering Institute (SEI) authorized lead appraiser from an external organization in the following Process Areas: [SWE-032]
a. Requirements Management.
b. Configuration Management.
c. Process and Product Quality Assurance.
d. Measurement and Analysis.
e. Project Planning.
f. Project Monitoring and Control.
g. Supplier Agreement Management.

Note: Organizations who have completed Standard CMMI® Appraisal Method for Process Improvement (SCAMPI) Class A appraisals against the CMMI® Model are expected to have their results posted on the SEI web site so that NASA can assess the current maturity state in the selection process.

2.5.2 The project shall assess options for acquisition against analysis of appropriate criteria to include risk, cost, and benefits for each option listed below: [SWE-033]
a. Acquire an off-the-shelf software product that satisfies the requirement.
b. Develop the software product or obtain the software service internally.
c. Develop the software product or obtain the software service through contract.
d. A combination of a, b, and c above.
e. Enhance an existing software product or service.

2.5.3 The project shall define and document or record the acceptance criteria and conditions for the software. [SWE-034]

2.5.4 For new contracts the project shall establish a procedure for software supplier selection including proposal evaluation criteria. [SWE-035]

2.5.5 The project shall determine which software processes, activities, and tasks are appropriate for the project. [SWE-036]

2.5.6 The project shall define the milestones at which the software supplier(s) progress will be reviewed and audited as a part of the monitoring of the acquisition. [SWE-037]

Note: All known contract milestones are expected to be included in the resulting contract.

2.5.7 The project shall document software acquisition planning decisions. [SWE-038]

Note: This may be in an acquisition plan or in another project planning document.

2.6 Software Contract Requirements

The requirements in this section are applicable for NASA contracted software procurements (e.g., reuse of existing software, modification of existing software, contracted and subcontracted software, and/or development of new software). Acquisition requirements are focused both inside the acquisition organization to ensure the acquisition is conducted effectively and outside the acquisition organization as the organization conducts project monitoring and control of its suppliers. These acquisition requirements provide a foundation for acquisition process discipline and rigor that enables product and service development to be repeatedly executed with high levels of ultimate acquisition success. This section contains the software acquisition requirements that should be performed by NASA organizations acquiring systems and/or services.

2.6.1 Government software insight requirements

2.6.1.1 The project shall require the software supplier(s) to provide insight into software development and test activities, including monitoring integration and verification adequacy, trade study data, auditing the software development process, and participation in all software reviews and technical interchange meetings. [SWE-039]

2.6.1.2 The project shall require the software supplier(s) to provide NASA all software products and software process tracking information, in electronic format, including all software development and management metrics. [SWE-040]

2.6.1.3 The project shall require the software supplier(s) to notify the project, in the response to the Request for

Proposals, as to whether open source software will be included in code developed for the project. [SWE-041]

2.6.1.4 The project shall require the software supplier(s) to provide NASA with electronic access to the source code developed for the project, including modified off-the-shelf software and non-flight software (ground test software, simulations, ground analysis software, ground control software, science data processing software, hardware manufacturing software, or other). [SWE-042]

Note: All known contract requirements should be addressed in the Request for Proposals and included in the resulting contract. Additionally, if the project needs to control further use and distribution of the resulting software or requires unlimited rights in the software, e.g., right to use, modify, and distribute the software for any purpose, the project should consider having the software copyright assigned to the government. This should be addressed in the Request for Proposals. The project should consult the Center Chief of Patent/Intellectual Property Counsel regarding required rights in the software.

2.6.2 Supplier Monitoring Requirements

2.6.2.1 The project shall require the software supplier to track all software changes and provide the data for the project's review. [SWE-043]

2.6.2.2 The project shall require the software supplier(s) to provide software metric data as defined in the project's Software Metrics Report. [SWE-044]

Note: The requirements for the content of the Software Metric Report are defined in Chapter 5.

2.6.2.3 The project shall participate in any joint NASA/contractor audits of the software development process and software configuration management process. [SWE-045]

2.6.2.4 The project shall require the software supplier(s) to provide a software schedule for the project's review and updates as requested. [SWE-046]

2.6.2.5 The project shall require the software supplier(s) to make available, electronically, the software traceability data for the project's review. [SWE-047]

2.6.2.6 The project shall document in the solicitation the software processes, activities, and tasks to be performed by the supplier. [SWE-048]

| TOC | Preface | Chapter1 | Chapter2 | Chapter3 | Chapter4 | Chapter5 | Chapter6 |
AppendixA | AppendixB | AppendixC | AppendixD | ALL |

| NODIS Library | Program Formulation(7000s) | Search |

DISTRIBUTION:
NODIS